

## TITLE OF THE INVENTION:

Data Collection in a Computer Cluster

## BACKGROUND OF THE INVENTION:

### Field of the Invention:

**[0001]** The present invention relates generally to computer clusters that include a plurality of computer nodes. More particularly, the present invention relates to a mechanism for collecting state information within the cluster. In this context, state information refers to data that indicates how the resources of a computer node are able to complete their tasks in the cluster. The state information may thus include, not only data indicating the current load of the various resources of a computer node, but also data about the current performance or capacity of the resources in the computer node, i.e. data about the current ability of the resources to complete their tasks in the cluster.

### Description of the Related Art:

**[0002]** As is commonly known, a computer cluster is a group of computers working together to complete one or more tasks. Computer clusters can be used for load balancing, for improved fault tolerance (i.e., for improved availability in case of failures), or for parallel computing, for example.

**[0003]** A typical computer cluster comprises a plurality of computer nodes. A computer node here refers to an entity provided with a dedicated processor, memory, and operating system, as well as with a network interface through which it can communicate with other computer nodes of the cluster. At least one of the computer nodes in the cluster is capable of acting as a manager node that manages the cluster.

In order to detect failures in the cluster, the manager node sends certain messages, called heartbeats, periodically to the other computer nodes in the cluster. Typically, only one computer node at a time acts as a manager node.

[0004] Control software, residing typically in the manager node, has to monitor all computer nodes that belong to the cluster. In order to get a true and up-to-date picture of the state of the nodes, the control software has to collect state information at a fairly high frequency from the nodes. This is a problem especially in large computer clusters, which may contain tens, or even hundreds of computer nodes. In these large computer clusters the data collection rate has to be compromised in favor of the performance of the network and the computer nodes, to ensure that the network does not become congested due to the data collection and that the performance of the computer nodes remains at an acceptable level despite the data collection performed. In other words, in large clusters the data collection rate has to be compromised in order not to degrade the performance of the network or the computer nodes excessively.

[0005] The objective of the present invention is to eliminate or alleviate this drawback.

#### SUMMARY OF THE INVENTION:

[0006] The invention seeks to bring about a novel mechanism for collecting state information from the computer nodes of a computer cluster. The invention seeks to provide a mechanism that does not require the collection rate of the state information to be compromised in favor of network or node performance even in large clusters.

[0007] In the present invention, an internal property of a computer cluster, the heartbeat mechanism, is utilized for collecting state information from the computer nodes for monitoring and control purposes. As described below, the collected state information may be utilized either internally in the computer cluster or by an outside entity, such as a network monitoring or management system.

[0008] According to one embodiment of the invention, a method for transferring state information in a computer cluster uses a plurality of computer nodes. The method includes the steps of transmitting a heartbeat message from a first computer node of a computer cluster to a second computer node of the computer cluster, where the second computer node includes at least one resource for performing at least one cluster-specific task and receiving the heartbeat message in the second computer node. The method also includes retrieving state information for a heartbeat acknowledgment message to be sent as a response to the heartbeat message, the state information indicating the ability of the at least one resource to perform the at least one cluster-specific task and sending the state information in the heartbeat acknowledgment message to the first computer node.

[0009] In another embodiment, the invention provides a computer cluster having a plurality of computer nodes. The computer cluster includes first means for transmitting a heartbeat message from a first computer node of the computer cluster to a second computer node of the computer cluster, where the second computer node includes at least one resource for performing at least one cluster-specific task, and second means for receiving the heartbeat message in the second computer node. The computer cluster also includes third means for retrieving state information for a

heartbeat acknowledgment message to be sent as a response to the heartbeat message, the state information indicating the ability of the at least one resource to perform the at least one cluster-specific task and fourth means for sending the state information in the heartbeat acknowledgment message to the first computer node.

**[0010]** In another embodiment, the invention provides a computer node for a computer cluster. The computer node includes at least one resource for performing at least one cluster-specific task, first means for receiving a heartbeat message from another computer node, second means for retrieving state information for a heartbeat acknowledgment message to be sent as a response to the heartbeat message, the state information indicating the ability of the at least one resource to perform the at least one cluster-specific task and third means, responsive to the second means, for sending the state information in the heartbeat acknowledgment message to the another computer node.

**[0011]** By means of the invention, real-time state information can be collected from the computer nodes of a computer cluster without excessively loading the network or the computer nodes, i.e. the information collection rate does not need to be compromised due to the load the collection causes. The overhead caused by the increased length of the acknowledgment message is relatively low, especially if the length of the minimum transmission unit is not exceeded.

**[0012]** In one embodiment of the invention, a computer node receiving a heartbeat message checks whether state information is to be retrieved for the heartbeat acknowledgment message to be sent as a response to the heartbeat message. In this way, unnecessary transfer of state information can be avoided.

**[0013]** A further advantage of the invention is that the collected information may be simultaneously utilized by different entities within or outside of the computer cluster.

**[0014]** Other features and advantages of the invention will become apparent through reference to the following detailed description and accompanying drawings.

#### **BRIEF DESCRIPTION OF THE DRAWINGS:**

**[0015]** In the following, the invention and its preferred embodiments are described more closely with reference to the examples shown in FIG. 1 to 5 in the appended drawings, wherein:

**[0016]** FIG. 1 illustrates one computer cluster according to the invention;

**[0017]** FIG. 2 is a flow diagram illustrating the basic operation of a manager node in view of one heartbeat message;

**[0018]** FIG. 3a is a flow diagram illustrating one embodiment for sending state information from a computer node;

**[0019]** FIG. 3b is a flow diagram illustrating another embodiment for sending state information from a computer node;

**[0020]** FIG. 4 is a schematic diagram illustrating the collection of state information in a computer node; and

**[0021]** FIG. 5 is a schematic presentation of a heartbeat message according to the invention.

#### **DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT(S):**

**[0022]** FIG. 1 shows an example of a computer cluster 100 in which the mechanism of the invention is utilized. The cluster comprises N computer nodes 110;

( $i=1,2,3,\dots,N$ ). Each computer node is an independent entity provided with a processor, memory and an operating system copy of its own. Each computer node is further provided with a network interface for connecting it to a network 120, which is typically an Internet Protocol (IP) based network. It is to be noted here that the mechanism of the invention is not dependent on the transmission protocol, but may be applied in many different environments. However, an IP network forms a typical environment for the invention.

**[0023]** At each time, one of the computer nodes, in this example node 110<sub>1</sub>, operates as a manager node that manages the cluster and its resources. In order to detect failures occurring in the cluster, the manager node sends heartbeat messages HB periodically to the other computer nodes in the cluster. Although the cluster may include more than one node being able to act as a manager node, one of such nodes operates as the manager node at a time. A single heartbeat message is typically a multicast message destined for all nodes of the cluster, and the period between two successive heartbeat messages depends greatly on the application environment.

**[0024]** When a computer node receives a heartbeat message from the manager node, it returns a heartbeat acknowledgment message HB\_ACK to the manager node, indicating to the manager node that it is alive and can therefore remain in the cluster. If the manager node does not receive a heartbeat acknowledgment message from a computer node, it starts recovery measures immediately. Typically, the computer node with which a communication failure has been detected is removed from the cluster, and the cluster-specific activities of the node are reassigned to one or more other nodes

**[0025]** A variety of different tasks may be performed by the cluster, and the actual applications may be distributed in a variety of ways within the cluster. One or more of the cluster nodes may appear as a single entity to an element external to the cluster. For example, if the computer nodes perform routing, one or more of the computer nodes may form a routing network element, as seen from the outside of the cluster. In another example, all computer nodes appear as a single entity to an external viewer.

**[0026]** If load sharing groups are utilized in the cluster, one or more of the computer nodes may further operate as an Internet Protocol Director (IPD) node, which is a load sharing control node routing incoming task requests within a load sharing group. In the example of FIG. 1, computer node 110<sub>2</sub> operates as an IPD node receiving task requests from the outside of the computer cluster.

**[0027]** In the present invention, the intrinsic heartbeat mechanism of a computer cluster is utilized for collecting state information from the computer nodes. The data may be collected for the purposes of the cluster only, or for an entity external to the cluster, such as a network monitoring or management system 160 connected to the network. The heartbeat acknowledgment messages are used to carry state information from the cluster nodes to the manager node, which then stores the information in a Management Information Base (MIB) 150.

**[0028]** In one embodiment of the invention, the MIB is made available for both entities within the computer cluster and for entities external to the computer cluster. For example, the internal fault management of the cluster may utilize the data collected. The fault management logic may be distributed in the cluster with an agent 130 residing in the manager node so that the fault management system can read data

from the MIB. In other words, the fault management system may comprise a client-server mechanism with the server part residing in the manager node and the client parts residing in the computer nodes. Another cluster entity capable of utilizing the MIB is a computer node that allocates incoming tasks to the computer nodes performing said tasks. In addition to the above-mentioned IPD node, any other cluster node may operate as such a load balancing entity.

[0029] Access to the MIB can be implemented in any known manner either directly or through the manager node, depending on whether the MIB forms an independent network node or whether it is connected to the manager node. The MIB may also be connected to a computer node other than the manager node.

[0030] FIG. 2 is a flow diagram illustrating an example of the basic operation of the manager node with respect to one heartbeat message sent to another computer node. It is thus to be noted here that FIG. 2 illustrates the operation with respect to one heartbeat message sent, i.e., the periodic sending of the heartbeat messages is not shown in the figure. When the manager node transmits a heartbeat message, it sets a timer (step 201) and starts to monitor if a heartbeat acknowledgment message is received as a response from said another computer node (step 202). If this acknowledgment message arrives before the expiration of the timer, the manager node examines the message (step 204). If the manager node detects the message contains state information, it extracts the said information from the message and updates the MIB based on the information (step 207). In case of an acknowledgment message void of state information the manager node proceeds in a conventional manner.

**[0031]** If the timer expires before a heartbeat acknowledgment message is received, the manager node concludes that a communication failure has occurred with the computer node, and starts recovery measures (step 205). In practice, the time period measured by the timer is so long that more than one heartbeat messages can be transmitted within that period. A heartbeat acknowledgment received for any of these messages then triggers the process to jump to step 204. Normally the manager node proclaims a computer node to be faulty when N successive heartbeat messages remain without an acknowledgment from that computer node. The manager node may thus be allowed to lose a given number of heartbeat messages before the recovery measures are started. Particularly in case of the UDP (User Datagram Protocol), which is commonly used for carrying heartbeat messages, messages may be lost without a real problem existing in the network. In view of the above, FIG. 2 is to be seen merely as an illustration of the processing principles of the incoming heartbeat acknowledgment messages in the manager node, while the actual implementation of the relevant manager node algorithm may vary in many ways.

**[0032]** FIG. 3a is a flow diagram illustrating an example of the operation of a computer node with respect to one heartbeat message received from the manager node.

When the heartbeat message is received, the computer node examines (step 301) whether a predetermined condition is fulfilled. This predetermined condition is set in order not to transfer state information unnecessarily in the acknowledgment messages.

If the condition is fulfilled, the computer node retrieves state information from its memory (step 303) and generates a heartbeat acknowledgment message containing the state information retrieved. If the predetermined condition is not fulfilled, the

computer node generates a normal heartbeat acknowledgment message, i.e. a heartbeat acknowledgment message without state information (302). The generated message is then sent back to the manager node (step 305).

**[0033]** The predetermined condition set for the retrieval of the state information is typically such that a certain minimum time period must have passed since the latest transmission of state information to the manager node. If this time limit has been exceeded, new state information is retrieved and inserted into the heartbeat acknowledgment message. Otherwise a normal heartbeat acknowledgment message is sent. In order to detect when the time limit has been exceeded, the computer node may start a counter at step 305. The current value of the counter is then examined at step 301 in connection with a subsequent heartbeat message. The computer node thus typically sends both normal heartbeat acknowledgment messages and heartbeat acknowledgment messages containing the state information, the proportions of these two message types depending on the rate of the heartbeat messages received.

**[0034]** The predetermined condition set for the retrieval of the state information may also consist of several sub-conditions that must be fulfilled before state information is retrieved. If the load of the computer node is used as such a sub-condition, the retrieval of the state information could occur, for example, only if both a certain minimum time period has passed since the latest transmission of state information and the current load of the computer node is below a certain maximum level.

**[0035]** As shown in FIG. 3b, it is also possible that the node determines, in response to the reception of a heartbeat message, the type of state information to be retrieved (step 311). Different types of information may thus be carried by successive heartbeat

acknowledgment messages. For example, if heartbeat messages are transmitted frequently enough, a certain set of parameters may be carried by N successive heartbeat acknowledgment messages, the same set being again transmitted by the next N heartbeat acknowledgment messages, and so on. Furthermore, certain information (parameters) may be transferred less frequently than other information.

[0036] The state information retrieved from the memory depends generally on the application running on the computer node. However, certain basic parameters that relate to the operating system of the computer node are the same for all computer nodes. These parameters include figures indicating the CPU idle time and the number of certain I/O operations, for example. Basically, the state information can be divided into two groups: the parameters relating to the performance of the applications and the parameters relating to the performance and/or state of the node platform.

[0037] FIG. 4 illustrates an example of the software architecture of the heartbeat acknowledgment generation in a computer node. A kernel module 400 residing in the kernel space receives the parameters relating to the operating system directly from the kernel space of the computer node. In the user-space, where the applications are executed, each application 401 may have a library 402 through which it can write the relevant parameters to the kernel module. A supervision agent 403 residing in the user space retrieves the state information from the kernel module if the predetermined condition is fulfilled, and constructs the heartbeat acknowledgment message containing the information retrieved. In the embodiment of FIG. 4, the storage of the state information is thus implemented in the operating system, which provides a faster

operation. However, the state information may also be stored in a mass memory, such as a disk.

**[0038]** FIG. 5 illustrates a general structure of the heartbeat acknowledgment message containing state information. The message comprises three successive portions: a header portion 501 that includes the protocol headers of the relevant protocols (such as Ethernet, IP and TCP/UDP headers), an acknowledgment identifier 502, and a payload portion 503 that contains the state information retrieved in the computer node. The message is thus otherwise similar to a conventional heartbeat acknowledgment message, but it includes a payload portion that contains the state information. In one embodiment of the invention the payload portion is encoded by using ASN.1 (Abstract Syntax Notation One) and PER (Packed Encoding Rules) coding. In this way the state information can be packed efficiently and more information can be inserted into the same message space. Depending on the protocols used, part of the state information may be transmitted without causing any extra load in the network. This is the case if the length of a conventional heartbeat message is shorter than the length of the minimum transmission unit, in which case state information may be used as the padding bits.

**[0039]** The load increase caused by a heartbeat acknowledgment message of the invention is relatively small as compared to the load caused by a conventional heartbeat acknowledgment message. This is because the overhead caused by a longer message is relatively low, since in short messages the protocol header takes a major part of the transmitted message. Furthermore, as messages shorter than a minimum message length are normally filled up, they may now be filled with the state

information. In this way part of the state information may be transferred without causing extra load in the network. The extra load caused by the method of the invention therefore also depends on the environment where the invention is applied. In an Ethernet network, for example, this minimum message length is 64 bytes, which is more than portions 501 and 502 require.

**[0040]** Although the invention was described above with reference to the examples shown in the appended drawings, it is obvious that the invention is not limited to these, but may be modified by those skilled in the art without departing from the scope and spirit of the invention. For example, it is not necessary to check whether a normal heartbeat acknowledgment message or a heartbeat acknowledgment message containing state information is to be sent, but an acknowledgment message containing state information can be sent in response to every heartbeat message.